A-1

ADAPTIVE MACHINE VISION
ANNUAL REPORT

ONR Contract N00014-86-C-0601

SAIC Science Applications International Co

ADAPTIVE MACHINE VISION
ANNUAL REPORT

ONR Contract N00014-86-C-0601

William W. Stoner
18 June 1992

SCIENCE APPLICATIONS INTERNATIONAL CORPORATION
6 Fortune Drive
Billerica, MA 01821
(508) 667-6365

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
Adaptive Machine Vision

5. TYPE OF REPORT & PERIOD COVERED
Progress Report
6/90 to 6/91

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
William W. Stoner

8. CONTRACT OR GRANT NUMBER(s)
N00014-86-C-0601

9. PERFORMING ORGANIZATION NAME AND ADDRESS
SCIENCE APPLICATIONS INTERNATIONAL CORPORATION
6 Fortune Drive
Billerica, MA 01821

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
Strategic Defense Initiative Organization
Innovative Science and Technology Office,
Washington, D. C.

12. REPORT DATE 6/18/92

13. NUMBER OF PAGES 22

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)
Office of Naval Research
800 Quincy Street
Code 1164
Arlington, VA 22217-5000

15. SECURITY CLASS. (of this report)
Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)
Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)
Machine Vision, Pattern Recognition

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Hierarchial, feature-based pattern recognition is implemented with a coherent optical correlator. The correlator uses a liquid crystal spatial light modulator at the input, Vander

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered) and changing the filters. The pattern recognition system is based upon the same principles as the Neocognitron developed by Fukushima. The system has potential use for SDI target/decoy discrimination. For testing purposes, simulated angle-angle and range-Doppler images were generated with SENSORSIM, a code developed and owned by SPARTA, Inc. This code uses constructive solid geometry to permit complex shapes to be built up from a small set of primitive shapes. Mathematics needed for ray tracing with constructive solid geometry is developed in Appendix 6. of this report.

TABLE OF CONTENTS

LIST OF TABLES AND FIGURES

<u>Tables</u> <u>Page</u>

<u>Figures</u>

1.0 Introduction

This annual report for the Adaptive Machine Vision contract covers the period from June 1990 to June 1991. During the period, a joint reseach effort called RADONN (for laser RAdar Discrimination with an Optical Neural Network) was initiated with Tein Hsin Chao of the Jet Propulsion Laboratory. SAIC's contributions to RADONN are reported here.

Work under the contract reported in March 1988 included analyses of the scaling properties and computational requirements of the Neocognitron, a pattern recognition neural network developed by Fukushima.[1] The RADONN effort builds upon this previous work. Our 1988 report also covered work on an architecture for coherent optical processing of stereo imagery, and simulations of an AC-coupled artificial retina which is adaptive to changes in light level. Our January 1989 report explored hardware concepts for the artificial retina. The 1989 report also treated the problem of tracking multiple targets simultaneously with a framing or scanning sensor. Candidate tracking algorithms were evaluated using receiver operating characteristic (ROC) curves. Because of a funding hiatus in 1989, this work was suspended, and with the resumption of funding in June 1990, all contractual activity has supported RADONN.

2. Statement of the Problem

To frustrate ballistic missile defenses, an attacker is likely to deploy decoys and chaff along with re-entry vehicles containing nuclear warheads. Since inflated decoys and chaff are lightweight compared to a Re-entry Vehicle (RV), the attacker does not suffer a significant reduction in "throw weight" by deploying several decoys per RV. Consequently, the SDI surveillance sensor must be capable of simultaneously tracking and discriminating a very large number of objects. An automatic system is needed to spot and hunt down the RVs, and this automatic system must be capable of distinguishing the RVs from the decoys.

The discrimination system may consist of an individual sensor or a mix of sensor types that jointly screen out the decoys. Because a mix of sensor types (see Table 2.1) may ultimately be required to unmask sophisticated decoys that are cleverly deployed, neural networks capable of fusing data from a mix of sensors are prime candidates for the discrimination system. Since the target signature cannot be known with certainty prior to an attack, the surveillance system must also be capable of adapting in real-time to the observed threat characteristics.

The Neocognitron uses a feature-fusion approach to discrimination. The evidence provided by individual image features is fused in stages, and at each stage, allowance is given to scale and aspect angle variations. As a result, the Neocognitron is able to robustly classify input images over a large range of scales and aspects. In contrast to the more popular "backprop" networks, the Neocognitron is able to adapt to changes in the target or background features without external supervision. The Neocognitron can adapt without supervision to changes in the targets or background clutter, because it uses a "competitive" learning mechanism. The potential for robust and adaptive target/background classification led to the selection of the Neocognitron architecture for the RADONN effort. However, in this initial phase of the effort, the hardware system is only intended to perform discrimination; there is no hardware implementation of adaption through competitive learning.

Table 2.1 Possible techniques for discrimination of RVs

Passive techniques

- angle-angle imaging of sunlit targets
- thermal imaging of targets in the earth's shadow

Active techniques

- laser target illumination of targets
  range-Doppler imaging
  angle-angle imaging
- neutral particle beam interrogation of targets

3.0 Approach

The SDI discrimination problem discussed above requires data fusion at high throughput rates. As illustrated in Figure

3.1, the Neocognitron provides a highly parallel architecture for performing data fusion. Within the Neocognitron, there are linear operations which can be carried out with optical processing, and nonlinear operations which are more suitable for electronic processing. This division of the processing is followed in the opto-electronic implementation proposed by Tein Hsin Chao which is illustrated in Figure 3.2.

An object is classified by first screening it for simple features (such as edges, or glints) with matched filter correlators, and then simplifying the resultant correlations. The simplifying is done reducing the spatial resolution and thresholding. The simplified output is then cycled through another stage of feature screening (with new matched filters) to find higher-level combinations of features. The output from this second stage of processing is also simplified by reducing the spatial resolution and thresholding, and recycled through a third stage, and so forth. Each additional stage of the screening increases the complexity of the feature combinations available for classification. At the same time, inconsequential image details are discarded by the simplification operations. Tolerance to scale and aspect angle is a consequence of the reduction in spatial resolution at each stage of the processing.

An oversimplified description of the hardware system is as follows. A liquid crystal light valve is used to input the image of the object to be classified. The input image is illuminated with spatially coherent light, and replicated with a Dammann grating,[2,3] so that the input may be simultaneously screened for different identifying features. The screening is performed with a bank of matched filters (optically or computer generated holographic filters of the VanderLugt type). The output of each VanderLugt correlator is simplified with a thresholding detector array. After one such pass through the optics, the simplified output becomes the input of the next processing stage. To hold down the hardware cost, this is accomplished by switching the VanderLugt filters after each pass so that all of the other hardware is reused in each processing stage.

3.10 RADONN Research Plan

We are following a crawl-walk-run plan. By adding hardware elements to the RADONN opto-electronic system as the components become available, problems are uncovered as soon as possible, and valuable hands-on experience is gained. The system now includes an Epson Crystal Light liquid crystal 320 by 220 input device, common Fourier transform lenses, a

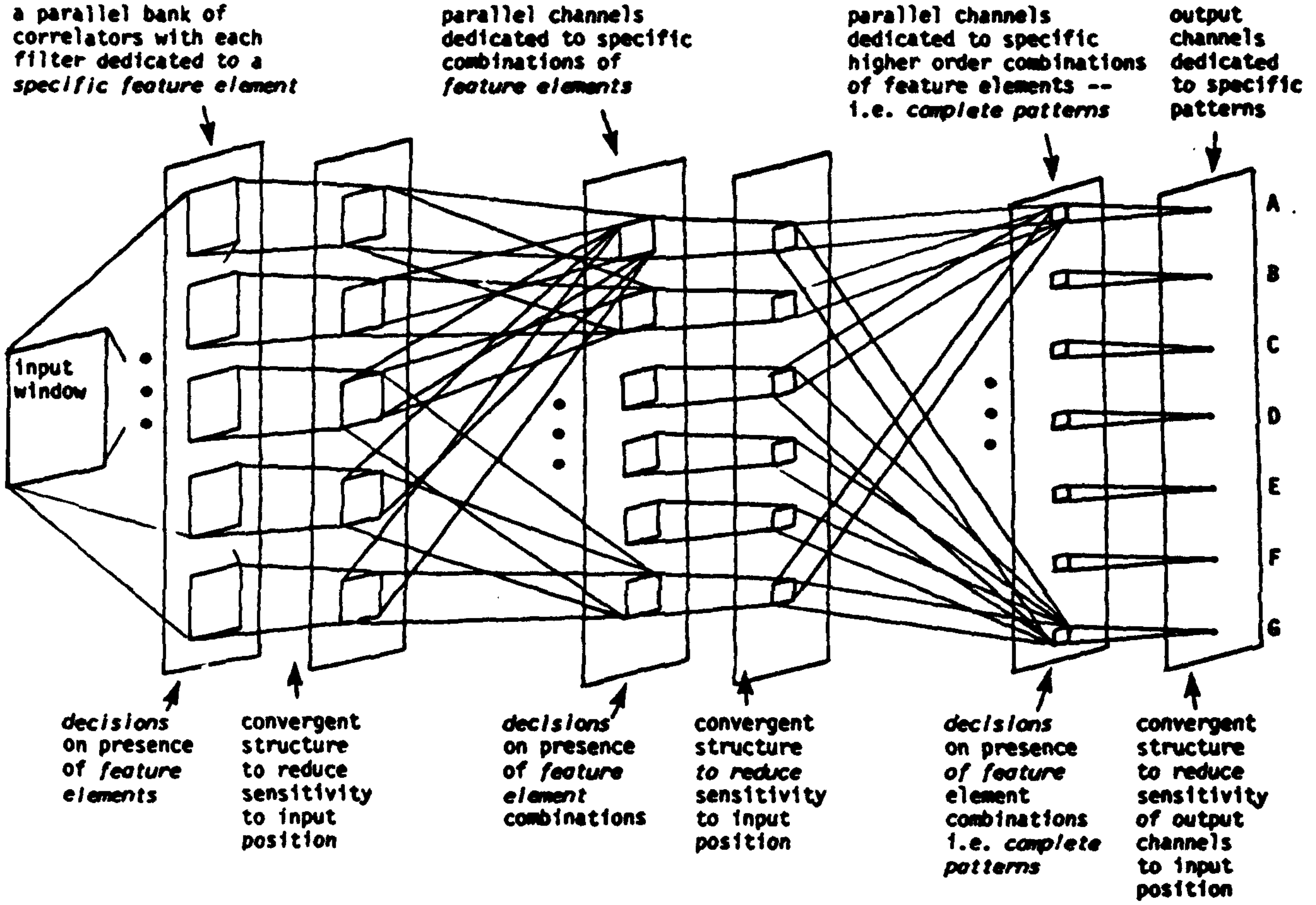
a parallel bank of correlators with each filter dedicated to a *specific feature element*
parallel channels dedicated to specific combinations of *feature elements*
parallel channels dedicated to specific higher order combinations of feature elements -- i.e. *complete patterns*
output channels dedicated to specific patterns
input window
A
B
C
D
E
F
G
decisions on presence of *feature elements*
convergent structure to reduce sensitivity to input position
decisions on presence of *feature element* combinations
convergent structure *to reduce* sensitivity to input position
decisions on presence *of feature* element combinations i.e. *complete patterns*
convergent structure to reduce sensitivity of output channels to input position

Figure 3.1 If we suppress the details in the Neocognitron architecture, we see that the processing is carried out by parallel data fusion channels that converge in a hierarchical structure to make decisions.

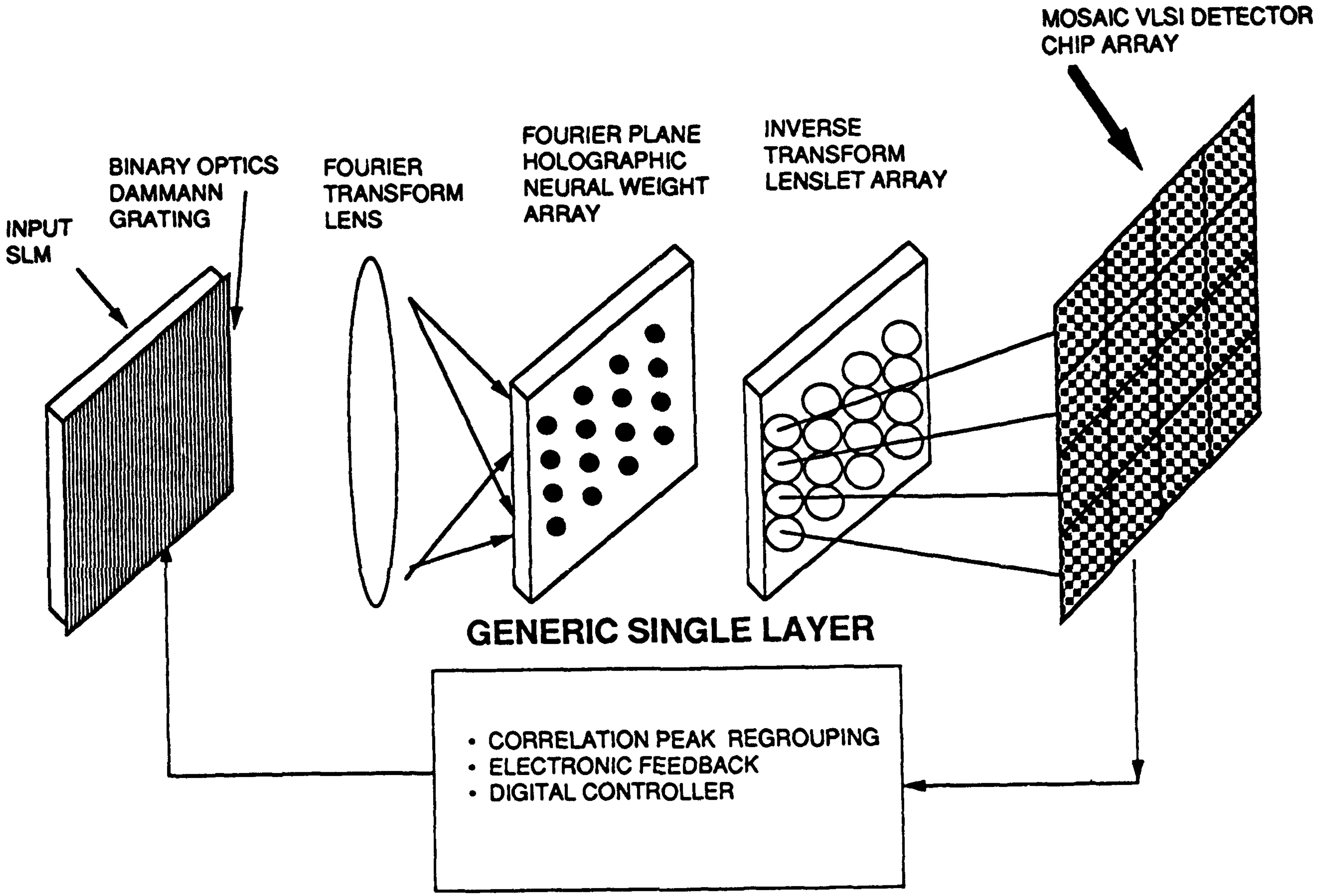
MOSAIC VLSI DETECTOR CHIP ARRAY
INPUT SLM
BINARY OPTICS DAMMANN GRATING
FOURIER TRANSFORM LENS
FOURIER PLANE HOLOGRAPHIC NEURAL WEIGHT ARRAY
INVERSE TRANSFORM LENSLET ARRAY
GENERIC SINGLE LAYER
• CORRELATION PEAK REGROUPING
• ELECTRONIC FEEDBACK
• DIGITAL CONTROLLER

Figure 3.2 Optoelectronic system modeled after the Neocognitron for distortion and shift invariant pattern recognition.

hololens array, and a 32 by 32 thresholding detector array. The threshold is variable, and set by an external rheostat. The chip was designed and delivered by the staff of the JPL Microfabrication Facility.

To exercise the system, 400 simulation images have been generated. The simulation images consist of 100 angle-angle cone images, 100 angle-angle beachball images, 100 range-Doppler beachball images, and 100 range-Doppler cone images. The simulation images were provided by SAIC, and this subject is discussed more thoroughly in Section 3.11. Representative images from this set of 400 are provided in Figures 3.3 through 3.6. Along with each image we have listed the scene description language used to drive the SENSORSIM code which rendered the image.

Next steps in the RADONN reseach plan include (1) generation of VanderLugt filters for feature screening, (2) fabrication of a Dammann grating, and (3) experiments with the thresholding detector.

To generate the VanderLugt filters, selected simulation scenes will be input on the Epson device, and an off-axis reference beam will be brought in to form a holographic filter on a high efficiency holographic material.

The Dammann grating is scheduled to be fabricated with the e-beam facility at JPL.

We have not determined whether or not a fixed threshold will work satifactorily with the Neocognitron architecture. However, it is possible to develop additional electronics to adapt the thresold. For example, the threshold could be selected to output a fixed number of pixels for the next stage of processing.

3.11 Generation of Simulation Images

We considered various alternatives for rendering simulation scenes, finally settling on SENSORSIM, a simulation code developed at SPARTA, Inc.[4] William J. Miceli of the Office of Naval Research brought SENSORSIM to our attention. Through a special arrangement with SPARTA, Inc., the Billerica office of SAIC was able to use SENSORSIM to simulate both conventional images (angle-angle images) and laser radar range-Doppler images for the RADONN effort.

The input to SENSORSIM consists of a Scene Description Language (SDL) specification of the object to be rendered. From the SDL input, SENSORSIM represents the object using the

```
 Title  BOOP0001
shape sphere -18045
  scale      .20000 1.0 1.0
  rotate z  135.00000
shape sphere -15046
  scale      .20000 1.0 1.0
  rotate z   90.00000
shape sphere -20048
  scale      .20000 1.0 1.0
  rotate z   45.00000
shape sphere -19041
  scale      .20000 1.0 1.0
  rotate z     .00000
combine intersection
combine intersection
combine intersection
scale 1.0 1.0     .20000
scale    2.96551   2.96551   2.96551
rotate y   23.58883
rotate z   21.55984
move 0.0    -.10000   -.01231
*  End of football.scn file
```

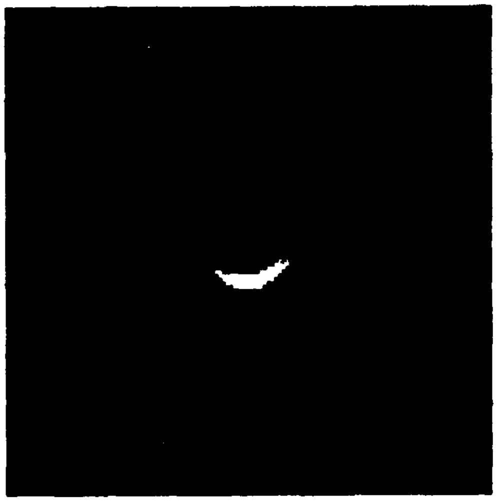

Figure 3.3 Angle angle image of beachball.

```
    Title  C00P0001
'  filename rv.scn
 shape cone -21047
      scale 0.581938004 0.581938004 2.171822198
 shape cylinder -21047
     move 0.0 0.0 1.0
     scale 0.601938004 0.601938004 0.987764572
     move 0.0 0.0 -0.02
 shape sphere -21047
     scale 0.06 0.06 0.06
     move 0.0 0.0 1.94
 combine union
 combine intersection
 shape cylinder -21047
     scale 0.601938004 0.601938004 0.106605103
     move 0.0 0.0 0.086605103
 shape sphere -21047
     scale 0.532277969 0.532277969 0.17742599
     move 0.0 0.0 0.17742599
 combine difference
 combine difference
 rotate y  174.86840
 rotate x  -43.72695
 move 0.0    1.48147    .60566
*  End of file rv.scn
```

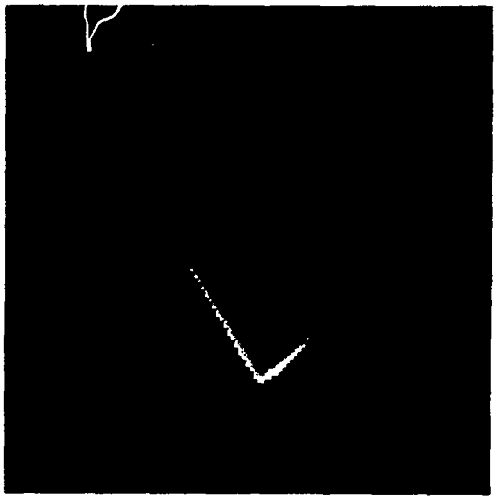

Figure 3.4 Angle angle image of cone.

```
ITLE__B00S0001
shape sphere -18045
  scale      .20000 1.0 1.0
  rotate z  135.00000
shape sphere -15046
  scale      .20000 1.0 1.0
  rotate z   90.00000
shape sphere -20048
  scale      .20000 1.0 1.0
  rotate z   45.00000
shape sphere -19041
  scale      .20000 1.0 1.0
  rotate z     .00000
combine intersection
combine intersection
combine intersection
scale 1.0 1.0     .20000
scale    2.96551   2.96551   2.96551
rotate y   23.58883
rotate z   21.55984
move     -.10000   -.01231    .07539
spin     -.10000   -.01231    .07539   3.30109   1.37663   8.40922
' End of football.scn file
```

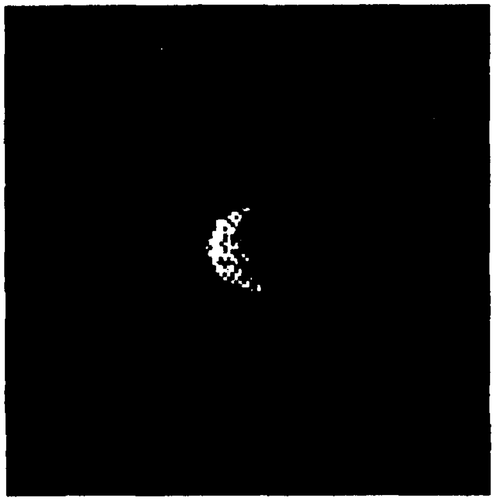

Figure 3.5 Range-Doppler image of beachball.

```
     Title  C00S0001
  filename rv.scn
shape cone -21047
     scale 0.581938004 0.581938004 2.171822198
shape cylinder -21047
    move 0.0 0.0 1.0
    scale 0.601938004 0.601938004 0.987764572
    move 0.0 0.0 -0.02
shape sphere -21047
    scale 0.06 0.06 0.06
    move 0.0 0.0 1.94
combine union
combine intersection
shape cylinder -21047
    scale 0.601938004 0.601938004 0.106605103
    move 0.0 0.0 0.086605103
shape sphere -21047
    scale 0.532277969 0.532277969 0.17742599
    move 0.0 0.0 0.17742599
combine difference
combine difference
rotate y  174.86840
rotate x  -43.72695
move    1.24657    .94515   2.33367
spin    1.33601    .25670   1.61392    .00357  -6.30024  -6.49206
  End of file rv.scn
```

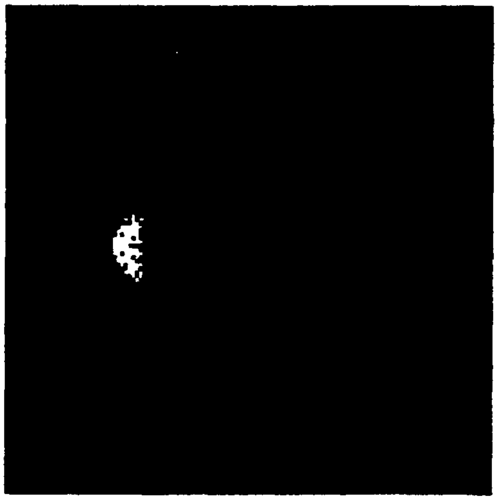

Figure 3.6 Range-Doppler image of cone.

technique of constructive solid geometry. In constructive solid geometry, complex shapes are built up from a small set of standard shapes. The standard shapes used by SENSORSIM are a sphere, a cone, a cube, and a torus. The standard shapes may be enlarged or shrunk, stretched or squeezed, translated or rotated. They may be combined to form new shapes by the Boolean set operations of union, intersection and subtraction.

Constructive solid geometry was originally developed for industrial applications of computer-aided design.[5,6] The technique is under current development by the model-based computer vision community.[7-12]

Since we needed to generate hundreds of simulations, we wrote routines to automatically generate SDL for two classes of objects: cones and beachballs. These routines generate randomized parameters for the cones and beachballs, so that the discrimination between the two classes cannot be made on the basis of a single feature (e.g. reflectivity or size). Table 3.1 identifies the parameters that were varied for the cones and beachballs.

Table 3.1 Cone and beachball parameters

	Is parameter varied?	
list of parameters	cone	beachball
specular reflectivity	yes	yes
diffuse reflectivity	yes	yes
aspect angle	yes	yes
shape	no	yes
size	no	yes
location 3-dimensional space	yes	yes
spin axis	yes	yes
spin rate	yes	yes

While the aspect angles of the beachballs were unrestricted, the cone aspects were randomly selected from a limited range. At its release point, an RV should be oriented so that it will not tumble upon re-entry. While in ballistic flight, the cone will maintain its orientation with respect to an inertial frame. At re-entry, aerodynamic forces control the cone's orientation. These considerations show that the solid angle of allowable aspect angles is governed by the extent of the (1) launch and (2) targeted areas in latitude and longitude. Consequently, we used a range of aspect angles for the cones that corresponds to launch sites in the Soviet Union and impact points within the continental United States.

In accordance with our crawl-walk-run philosophy, the scenes developed so far contain only a single object. Future scenes will contain multiple objects. The routine that generates SDL for scenes containing multiple objects should be capable of placing the objects in random locations within 3-dimensional space. In order to rule out nonsensical scenes in which the objects overlap in space, it is necessary to track the boundaries of each object as it is scaled, rotated, and translated.

We reviewed the literature of constructive solid geometric modeling (references 5-12) but were unable to find a treatment of this problem. Consequently, we developed mathematics to represent surfaces of objects generated with constructive solid geometry. The mathematics we developed gives the surface locations and surface orientations. This mathematics is provided in Section 6.0, Appendix on Ray Tracing for Constructive Solid Geometry.

4.0 List of Publications and Presentations

T. H. Chao, H. Langenbacher, S. Rosenzweig, and W. Stoner, RAdar Discrimination with an Optical Neural Network (RADONN)

(Poster paper at the Gordon Research Conference on Holography and Information Processing, 17-21 June 1991, Plymouth State College, Plymouth, New Hampshire.)

5.0 References

1. Kunihiko Fukushima, Sei Miyake, and Takayuki Ito, "*Neocognitron:* A neural Network Model for a Mechanism of Visual Pattern Recognition," IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-13, No. 5, September/October 1983, pp 826-834.

2. H. Dammann and E. Klotz, "Coherent Optical Generation and Inspection of Two-Dimensional Periodic Structures," Optica Acta vol. 24, p 505 (1977).

3. J. Jahns, M. E. Prise, M. M. Downs, S. J. Walker, N. Streibl "Dammann gratings as array generators," abstract of talk before the Annual Meeting of the Optical Society of America, Journal of the Optical Society of America, A, vol. 4, No. 13, p 69, (Dec. 1987).

4. SENSORSIM was developed by Mr. Norman R. Guivens Jr. of SPARTA, Inc., 24 Hartwell Ave., Lexington, MA 02173.

5. H. B. Voelcker and A. A. G. Requicha, "Geometric modeling of mechanical parts and processes," Computer, Vol. 10, Dec. 1977, pages 48-57.

6. H. B. Voelcker and Staff of the Production Automation Project, "The PADL-1.0/2 system for defining and displaying solid objects," Computer Graphics, Vol. 12, No. 3, Aug. 1978, pages 257-263.

7. Dana H. Ballard and Christopher M. Brown, Computer Vision, Chapter 9, "Representations of Three-Dimensional Structures," (1982, Prentice-Hall, Inc. Englewood Cliffs, NJ).

8. Thomas O. Binford, "Image understanding: intelligent systems," Proceedings: Image Understanding Workshop, February 1987, pages 18-31.

9. Jean Ponce and David Chelberg, "Localized intersections computation for solid modelling with straight homogeneous generalized cylinders," Proceedings: Image Understanding Workshop, February 1987, pages 933-941.

10. Thomas O. Binford, "Image understanding: intelligent systems," Proceedings: Image Understanding Workshop, April 1988, pages 17-28.

11. Jean Ponce and Glenn Healey, "Using Generic Geometric And Physical Models For Representating Solids," Proceedings: Image Understanding Workshop, April 1988, pages 488-503.

12. Thomas O. Binford, "Spatial understanding: the SUCCESSOR system," Proceedings: Image Understanding Workshop, May 1989, pages 12-20.

6.0 Appendix on
Ray tracing with Constructive Solid Geometry

Constructive Solid Geometry (CSG) was developed for computer aided design and display of complex shapes. The idea of CSG is to construct complex shapes out of simple shapes such as spheres, cubes, cones, and tori. The simple shapes are called "primitives."

The allowed constructions include combining operations (set union, set difference, and set intersection), and linear coordinate transformations (isotropic or anisotropic scaling, shearing, rotation, and translation). The concept is illustrated in Figure 1.

The primitives are given reflectance properties so that ray tracing may be used to render images of the constructions. Although a number of CSG computer programs have been reported in the literature, references 1-8, none of these reports provide very much detail on ray tracing for CSG.

Ray tracing requires: 1) coordinates of the surface and 2) the normal vectors of the surface in a reference system. The reference system remains fixed in space during the construction operations.

In addition to the reference coordinate system, body coordinate systems are needed. A body coordinate system is fixed to each primitive. All scaling, shearing, rotating, and translating operations apply to both the primitive and the body coordinates, so the equation of a primitive surface is invariant when expressed in body coordinates. The sequence of transformations applied to a primitive is used to derive the coordinate transform between the body and the reference systems.

Suppose the matrix M relates reference coordinates $\underline{x}_{ref}$ to body coordinates $\underline{x}_{body}$:

$$\underline{x}_{body} = M\underline{x}_{ref}.$$

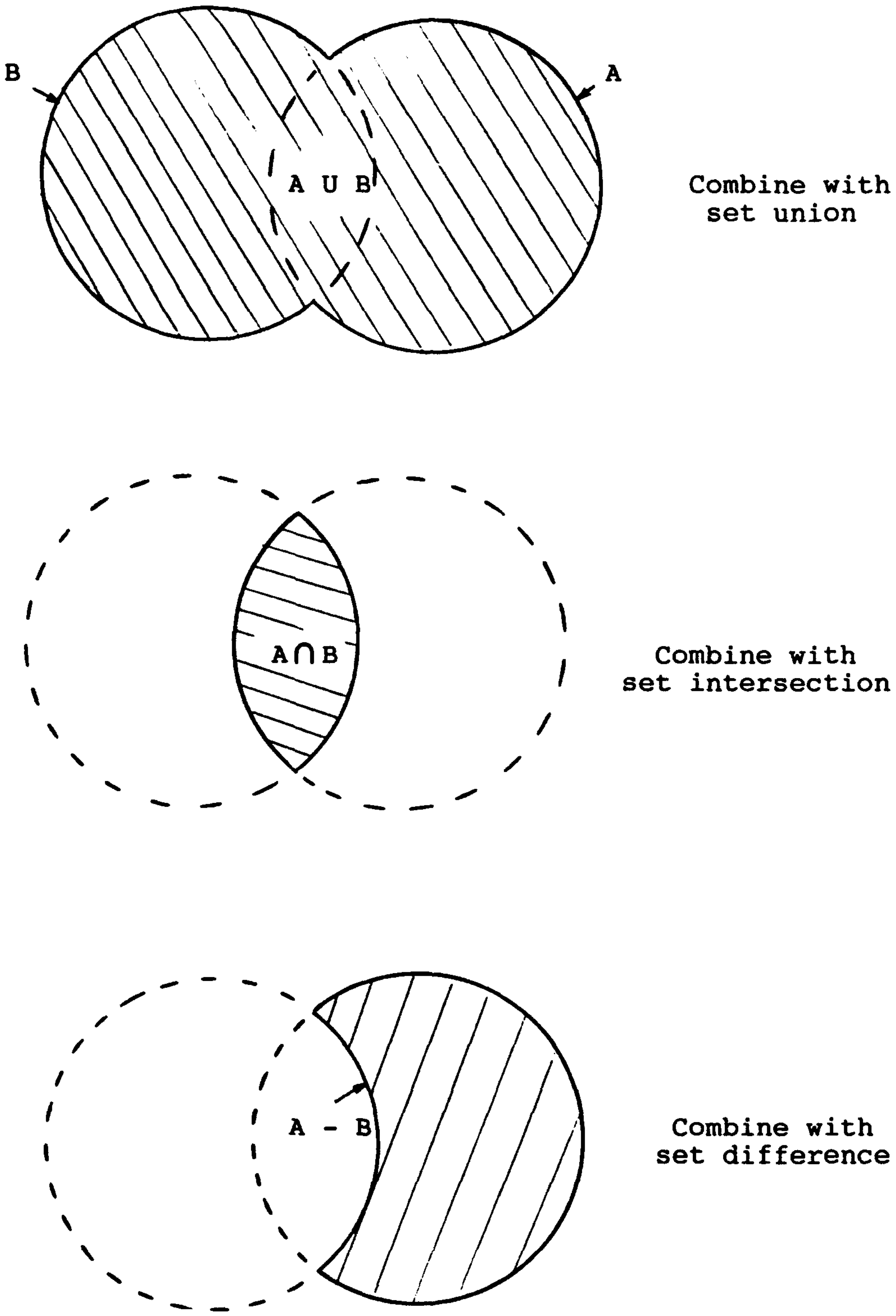

Figure 1, Part A: Combining operations

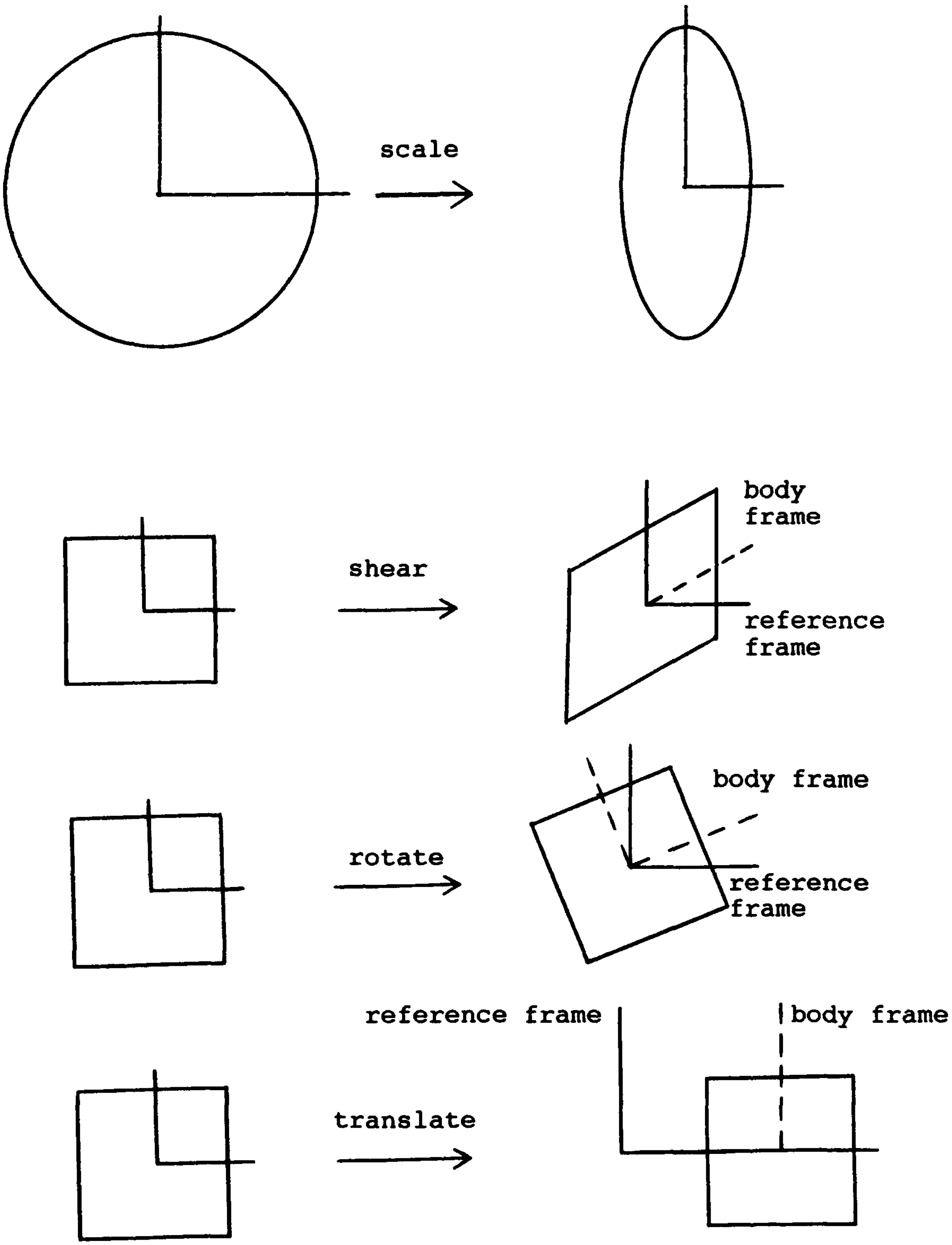

Figure 1, Part B: Linear coordinate transformations

Then, if the equation of a primitive surface in body coordinates is given by

$$F(\underline{x}_{body}) = 0,$$

the equation of the surface in the reference frame is

$$F(M\underline{x}_{ref}) = 0.$$

This transformed equation carries the information needed for ray tracing: the location and orientation of the surface after it has been scaled, sheared, rotated, and translated. Because the equation is expressed in the reference coordinates, solutions to the equation are the locations of surface points with respect to the reference frame. The orientation of the surface with respect to the reference frame is obtained from the gradient of $F(M\underline{x}_{ref})$; the chain rule is used to carry out the differentiation with respect to the reference coordinates $\underline{x}_{ref}$.

Recall that the gradient of a function is orthogonal to the surface over which the function has a constant value. This can be seen from the Taylor series expansion of $F(\underline{x})$:

$$F(\underline{x} + \underline{e}) = F(\underline{x}) + (\text{grad } F(\underline{x}))^t\underline{e} + \text{higher order terms}$$

If the incremental vector $\underline{e}$ lies on a surface for which the function is constant, this constant may be subtracted from the left and right of the equation. Then, in the limit as $\underline{e}$ approaches zero, the higher order terms become negligible and the equation reads:

$$(\text{grad } F(\underline{x}))^t\underline{e} = 0.$$

This can be recognized as the inner product (or dot product) of the gradient of F and the surface vector $\underline{e}$. Since the inner product is zero for all incremental surface vectors $\underline{e}$ at the surface point $\underline{x}$, the gradient of F must point along the normal to the surface. This result does not depend upon the particular coordinate system one chooses; $(\text{grad } F(\underline{x}))^t\underline{e}$ is an <u>invariant</u>.

To complete the programme for finding the location and orientation of surfaces constructed with CSG, it is necessary to:

1) represent the transformations of scaling, shearing, rotation, and translation with matrices, and

2) compute grad $F(M\underline{x}_{ref})$ with respect to the reference coordinates.

Scaling, shearing, and rotating can be represented by matrix multiplication. The diagonal matrix elements carry out scaling; the off-diagonal elements carry out shearing and rotating.

Translation is additive in nature, and it does not appear to have a matrix representation. However, shear is a translation depending upon two directions. One direction gives the orientation of the translation, the other direction gives the size of the translation. If a 4th dimension is introduced, it is possible to use shear for translation along the x, y, or z directions, with the amount of translation growing in proportion to the 4th coordinate. We are free to set this 4th coordinate to a constant, so that we get a translation in (x,y,z) and not a shear.

(In projective geometry, a system of 4-dimensional "homogeneous" coordinates is introduced to serve a similar purpose.)

We set the 4-th component equal to 1 so that the translations are scaled by unity:

$$v = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Translations of (x,y,z) are carried out with the 4 by 4 matrix T:

$$T = \begin{bmatrix} 1 & 0 & 0 & -xo \\ 0 & 1 & 0 & -yo \\ 0 & 0 & 1 & -zo \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The product of T with v is:

$$T \cdot v = \begin{bmatrix} x - xo \\ y - yo \\ z - zo \\ 1 \end{bmatrix}$$

The ordinary 3 by 3 matrices for scaling, shearing, and rotating may be embedded into 4 by 4 matrices so that all of the transformation matrices have the same dimensions. For example, to stretch a primitive along the x axis by a factor of 5, so that one unit along the body x axis corresponds to 5 units along the reference x axis, the following matrix is needed:

$$S = \begin{bmatrix} \frac{1}{5} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Similarly, to shear the body along the x axis in proportion to location along the z axis, a matrix of the following form is used:

$$D = \begin{bmatrix} 1 & 0 & -k & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Similarly, to rotate the body about the z axis by an angle ϕ, the following matrix is used:

$$R = \begin{bmatrix} \cos(\phi) & \sin(\phi) & 0 & 0 \\ -\sin(\phi) & \cos(\phi) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

A sequence of scaling, shearing, rotating, and translating operations are represented by an ordered matrix product of the individual operations. Here M represents to sequence:

O_1 first, O_2 second, O_3 third, O_4 last.

$$M = O_1 \cdot O_2 \cdot O_3 \cdot O_4$$

Since the function F defining the surface of the primitive is a function of the 3-vector for the point (x,y,z), the 4th component of the 4-vector should be stripped off. This is done with a projection matrix:

$$\text{P4to3} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

The P4to3 projection matrix allows 4-vectors to be used with the function that specifies the surface of a primitive shape. With the understanding that $\underline{x}_{ref}$ is a 4-vector, we can specify a surface transformed by M with the equation:

$$F(\text{P4to3}\cdot M\underline{x}_{ref}) = 0.$$

This notation is cumbersome, and for ease of reading, we shall suppress the projection matrix P4to3 when it occurs in the argument of F.

The gradient of F with respect to the reference coordinates turns out to be:

$$\text{P4to3}\cdot M^t \cdot \text{grad } F \Big| \text{ evaluated at } (M\underline{x}_{ref})$$

The factor of M^t in front of the gradient operator indicates that the surface normals transform differently than vectors on the surface. Surface normals transform differently because if they transformed in the same way as position vectors, the angle between the surface normal and the surface would change during stretching and shearing.

Another way to think about this is to use the invariant:

$$(\text{grad } F(\underline{x}_{body}))^{t}\underline{e}_{body}$$

This inner product is zero in all coordinate frames. In the reference frame, the gradient transforms to:

$$\text{P4to3}\cdot M^{t}\cdot\text{grad } F(\underline{x}_{body}),$$

and in terms of the reference coordinates the surface vector $\underline{e}_{body}$ is:

$$\text{P4to3}\cdot M^{-1}\underline{e}_{body}.$$

The inner product is therefore:

$$(\text{grad } F(\underline{x}_{body}))^{t}\cdot M(\text{P4to3}^{t}\cdot\text{P4to3})M^{-1}\underline{e}_{body}$$

The inner product of P4to3 with itself forms the 4 by 4 unit matrix, leaving:

$$(\text{grad } F(\underline{x}_{body}))^{t}\cdot MM^{-1}\underline{e}_{body}$$

The product of M and M^{-1} forms the 4 by 4 unit matrix, leaving the original expression:

$$(\text{grad } F(\underline{x}_{body}))^{t}\underline{e}_{body}$$

(Here the grad is a 4-dimensional vector, as is $\underline{e}_{body}$.

However, the 4th component of the grad is zero, because F is not a function of this component. So it turns out ok to treat both vectors as 3-vectors.)

The invariant property of the inner product of a surface vector and a surface normal results in different transformation laws for these two different types of vectors. In tensor analysis, this behavior leads to the distinction between covarient and contravarient vectors.

References

1) H. B. Voelcker and A. A. G. Requicha, "Geometric modeling of mechanical parts and processes," Computer, Vol. 10, Dec. 1977, pages 48-57.

2) H. B. Voelcker and Staff of the Production Automation Project, "The PADL-1.0/2 system for defining and displaying solid objects," Computer Graphics, Vol. 12, No. 3, Aug. 1978, pages 257-263.

3) Dana H. Ballard and Christopher M. Brown, Computer Vision, Chapter 9, "Representations of Three-Dimensional Structures," (1982, Prentice-Hall, Inc. Englewood Cliffs, NJ).

4) Thomas O. Binford, "Image understanding: intelligent systems," Proceedings: Image Understanding Workshop, February 1987, pages 18-31.

5) Jean Ponce and David Chelberg, "Localized intersections computation for solid modelling with straight homogeneous generalized cylinders," Proceedings: Image Understanding Workshop, February 1987, pages 933-941.

6) Thomas O. Binford, "Image understanding: intelligent systems," Proceedings: Image Understanding Workshop, April 1988, pages 17-28.

7) Jean Ponce and Glenn Healey, "Using Generic Geometric And Physical Models For Representating Solids," Proceedings: Image Understanding Workshop, April 1988, pages 488-503.

8) Thomas O. Binford, "Spatial understanding: the SUCCESSOR system," Proceedings: Image Understanding Workshop, May 1989, pages 12-20.